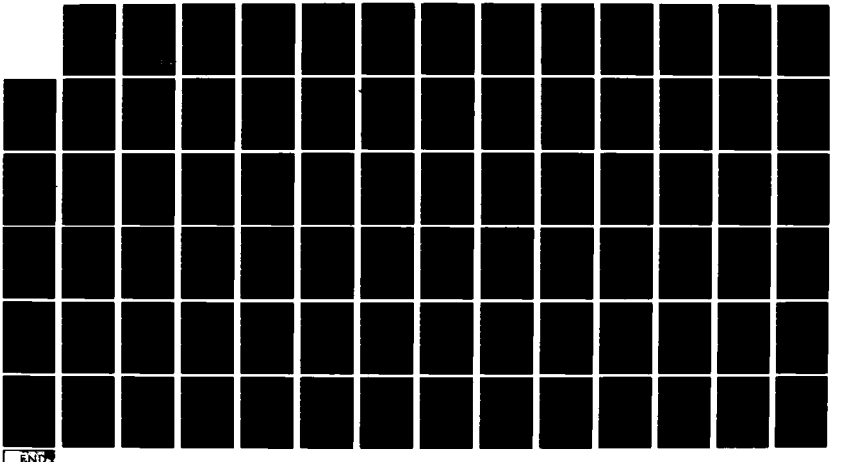
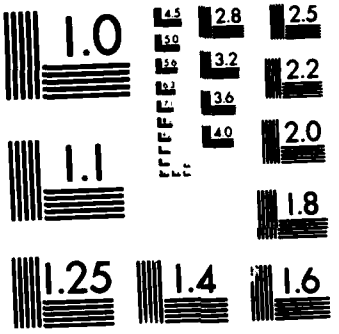
AD-A124 782 IMPLEMENTATION OF SOFTWARE FOR A MICROCOMPUTER BASED MOSSBAUER SPECTROMETER(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING B E PATE 1/1

UNCLASSIFIED MAR 82 AFIT/GNE/PH/82M-11 F/G 9/2 NL



END
FILMED
DTIC



AFIT/GNE/PH/82M-11

IMPLEMENTATION OF SOFTWARE FOR A MICROCOMPUTER BASED MOSSBAUER SPECTROMETER

Thesis

Bruce E. Pate
AFIT/GNE/PH/82M-11 Captain USA

Approved for public release; distribution unlimited.

AFIT/GNE/PH/82M-11

IMPLEMENTATION OF SOFTWARE FOR A MICROCOMPUTER BASED MOSSBAUER SPECTROMETER

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirement for the Degree of
Master of Science

by

Bruce E. Pate, B.S.
Captain USA

Graduate Nuclear Engineering

March 1982

Preface

This thesis is the product of my efforts to program an PDP-11 microcomputer to analyze the spectra generated by a Mossbauer spectrometer, so that a more responsive, interactive, and versatile evaluation can be made. This thsis will describe the program used to evaluate the spectra, the modifications made to the program to better work with the current spectrometer, the results of test evaluations made using absorbers of potassium ferrocyanide, natural iron, sodium nitroprusside, and neodymium cobalt, and how these results compared with the previous computer program and with published standard values.

I would like to thank Dan Zambon of the Electrical Engineering Department, whose assistance in the initial set up of the computer was invaluable. I especially want to thank my advisor, Dr. George John, whose guiding hand and support during this project helped make this thesis possible.

I also want to thank my wife, Kathy, whose moral support and understanding were always present when I needed them.

Bruce E. Pate

Contents

List of Figures

Abstract

Computer analysis of Mossbauer spectra using a microcomputer requires tight controls on memory requirements and programming techniques. In this study, a FORTRAN IV program, MOSFUN, was programmed into an PDP-11 microcomputer. The program was modified to calculate the velocity for a data point using the least squares fit of counts from a Moire interferometer and to perform an alphanumeric plot of the original data and the theoretical fitted spectrum. Spectra were obtained for absorbers of potassium ferrocyanide, natural iron, sodium nitroprusside, and neodymium cobalt. These spectra were then analyzed with the MOSFUN program and the results compared with analyses done with the former program, GENFIT, and with published values. Results showed MOSFUN to be more flexible, easier to adapt, and easier to control during the evaluation of spectra, while still providing results that agreed within 1% of those obtained with GENFIT and with standard values.

IMPLEMENTATION OF SOFTWARE FOR A MICROCOMPUTER BASED MOSSBAUER SPECTROMETER

I. Introduction

Problem

The ultimate purpose of this project is the design, construction, and testing of a microcomputer-based system capable of collecting, storing, and analyzing data from a Mossbauer spectrometer. This new system will replace an aging and unreliable multichannel analyzer (MCA) based system and will preclude the requirement to do analysis on a separate mainframe computer. This new system will allow data to be collected more efficiently and will provide more productive and timely analysis.

The objectives of this thesis were to program an PDP 11 microcomputer to analyze Mossbauer spectra and to assist in the design of new hardware to replace the current MCA. The software will allow more control over the fit of the Mossbauer spectra, since it will be interactive instead of the batch process of the current program. Analyses can be done more quickly and frequently since the computer is dedicated to the spectrometer. The new hardware is designed to provide the signals needed by the Mossbauer control unit to drive the velocity transducer, to multiplex the incoming signals, and to store the data until requested by the computer.

Background

Until recently, Mossbauer spectrometers have been based on systems which used multichannel analyzers (MCA) for acquiring and storing spectra. These systems required the data to be collected and then transferred in some way to a computer for analysis of the data. The transfer can be accomplished in a variety of ways, which vary in speed and tedium. The most rapid method would be a direct transfer of the data stored in the MCA memory to the memory of a computer. This may be by a direct connection to a minicomputer or over phone lines to a large centrally-based time-shared computer. An intermediate, but more time-consuming method, would be to transfer the data to magnetic tape or paper tape before transferring to a computer. In the case of paper tape and computer systems with inadequate input directly from paper tape, another step requiring the transfer of the data to punched cards may be required. The later example with paper tape has been the situation with the Mossbauer system and the facilities for analysis available at this laboratory.

An attractive alternative to the foregoing is the use of a microcomputer directly interfaced to a Mossbauer transducer and driver. This alternative has been enhanced by hardware and software which in recent years have become available in large variety and at moderate cost. In addition, programs for analysis of Mossbauer spectra have become available for a nominal fee. One such program, called MOSFUN, has been written in FORTRAN IV by E.W. Muller (Ref 3:1.1).

Theory

Mossbauer spectroscopy makes use of the Mossbauer effect, which was discovered by Rudolf Mossbauer in 1957 (Ref 2:12). By moving a radioactive source, Mossbauer discovered that the energy of the emitted photon was changed by a small amount. The energy was increased for a positive velocity and decreased for a negative one. This effect is used for spectroscopy by placing an absorber between the moving source and a detector. The absorber to be studied must contain the same atom as the source but in the ground state. If the energy of a photon exactly matches the energy needed to raise a nucleus in the absorber to a higher state, the photon may be absorbed. The number of counts seen by a detector will be reduced whenever the energy of a photon emitted by the source is changed enough to equal this resonance absorption energy. A plot of the number of counts recorded by the detector versus the velocity of the source results in a spectrum such as the one in Figure 1. The dips in the spectrum correspond to the excited energy levels in the absorber nuclei. The velocity of the source can be related to the energy of the absorbed photon and, from these energies, a great deal can be told about the states of the nuclei in the absorber. Each dip in the spectrum may be one or more overlapping energies. Each energy is called a line and each line will have a common shape. The shape is a convolution of the energy distributions from the source and the absorber. This distribution will be

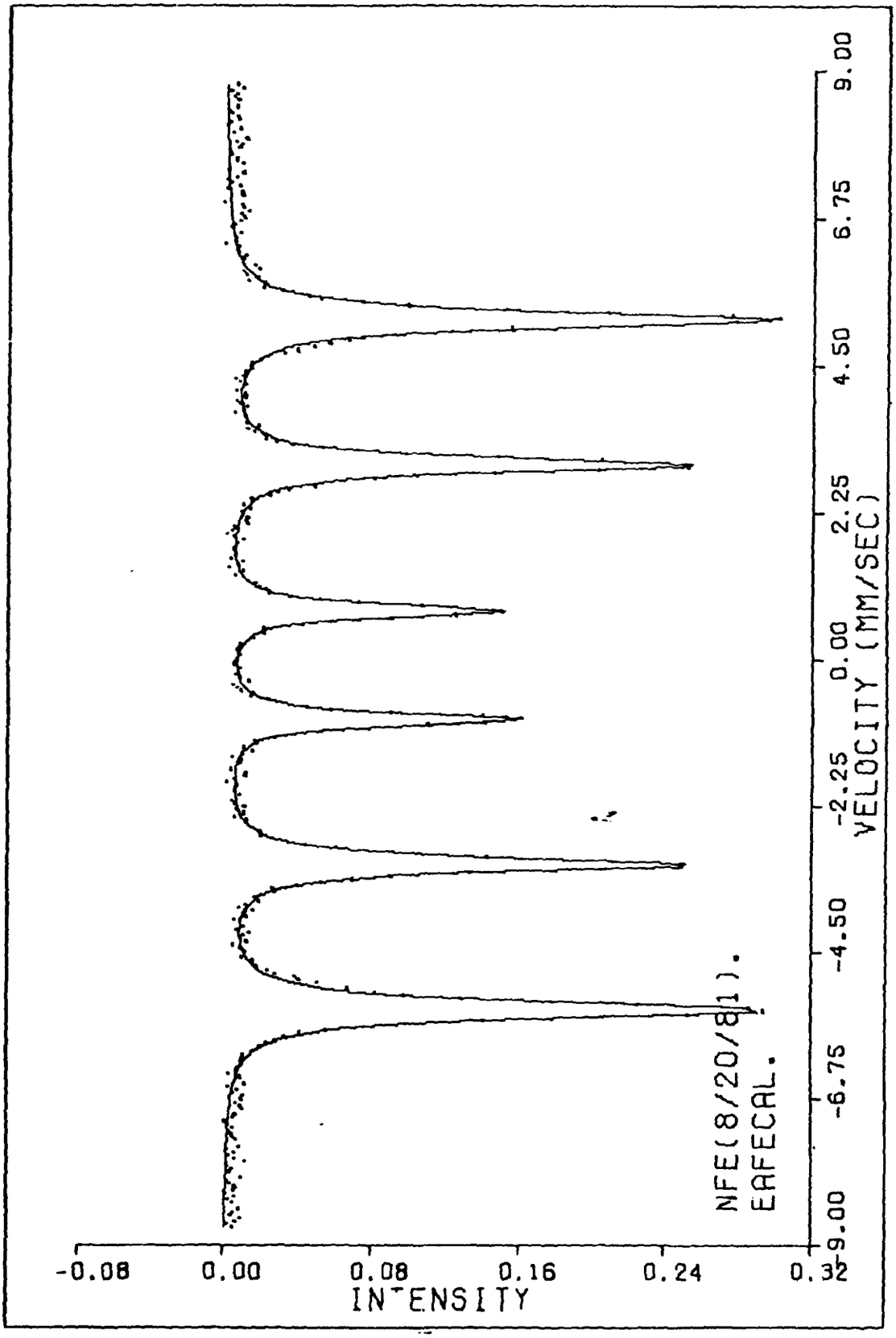


Figure 1. Example of a Mossbauer Spectrum

Lorentzian unless perturbed by the system. Detailed descriptions of Mossbauer theory and its application to spectroscopy can be found in the thesis by Towery (Ref 11:6).

A computer program to analyze this spectrum must be able to apply a Lorentzian distribution to each line, determine the intensity or magnitude of each dip, and locate the velocity of the lines using several known characteristics or parameters of the nuclear energy levels. The program then varies the parameters until it finds a set of theoretical lines, whose combined shapes and locations most nearly match the experimental data.

Scope

To interface the microcomputer to the Mossbauer system requires interface hardware to be built and tested to replace the functions currently done by the MCA. This new device must be interfaced with the PDP 11 microcomputer and the computer programmed to use the interface as an extension of its own memory. The new device has not been completed and these parts of the project will not be covered in this report.

Once in the computer, the data must be analyzed. This study is primarily concerned with the computer programming needed to analyze Mossbauer spectra after the data is in the computer. Chapter II discusses the equipment used in the spectrometer and the computer system used to evaluate the spectra produced. Chapter III discusses the parameters used to evaluate Mossbauer spectra, the computer program

MOSFUN and how it uses the parameters to evaluate spectra. Chapter IV discusses the results of experiments done with natural iron, sodium nitroprusside, potassium ferrocyanide, and neodymium cobalt, and how these results obtained using MOSFUN compared with results obtained with the current program, GENFIT, and with published standards. Chapter V discusses the conclusions and recommendations for this study.

II. Equipment Used

Mossbauer Equipment

In addition to the source and absorber, the components of the Mossbauer spectrometer are a velocity transducer (motor), a motor control unit (MCU) to drive the motor at a constant acceleration, a linear amplifier/single channel analyzer, a krypton-filled proportional counter, and a high voltage power supply for the detector. All except the detector were manufactured by Ranger Electronics. The MCU is driven by the channel-address ramp from an RIDL multichannel analyzer (Model 34-12B) driven in the time-sequential scaling mode by an RIDL time-base oscillator (Model 54-6). The equipment is the same as that used by Beller (Ref 1:10), and described in detail in the thesis by Skluzacek (Ref 8:5-9), and will not be repeated here.

The transducer can drive the source in either of two drive modes, triangular or flyback. The maximum velocity (VMAX) of the source is set on the MCU and the source is moved at a constant linear acceleration between +VMAX and -VMAX. The triangular drive mode is a full period drive mode and, as such, moves the source from +VMAX to -VMAX and back to +VMAX. It continues to cycle the source through this full period as counts from the detector accumulate in the MCA. Each pass between +VMAX and -VMAX

generates a complete spectrum such as the one in Figure 1. As a result, the full period drive mode generates two spectra, one from +VMAX to -VMAX and a mirror image spectrum from -VMAX to +VMAX. The flyback mode on the other hand is a half period drive mode. The source velocity is only varied from -VMAX to +VMAX and generates only a single spectrum rather than the double spectrum of a full period drive mode. The spectrum in Figure 1 is a half period spectrum (flyback) and is identical to the right half of a full period spectrum such as a triangular drive mode would create. The flyback drive mode was used exclusively in this study, but references will be made to programming considerations for full period spectra.

Source and Absorbers

The source used for obtaining spectra used in this study is 50 mCi cobalt-57 (date: 5 June 1981) diffused into a 6 micron thick rhodium matrix. The source is prepared by Amersham-Searle. The cobalt-57 decays into an excited state of iron-57, which in turn decays emitting an 14.4 kilo-electronvolt (keV) gamma ray among other radiations. This 14.4 keV gamma ray is the photon of interest for this study. Therefore, the absorber used must also contain iron-57 in the ground state that can be excited with a photon of this energy. Absorbers used in this study are a 25 micron thick natural iron foil, a National Bureau of Standards sodium nitroprusside crystal, a potassium ferrocyanide crystal, and a neodymium cobalt sample.

Computer System

The computer system consists of a PDP 11/03 microcomputer, an RX-02 dual 8-inch floppy disk drive, and an VT-100 terminal all manufactured by Digital Equipment Corporation. The memory is a 64 kilobyte CI-1103 memory board made by Chrislan Industries. An MX-80F/T dot matrix printer by Epson is connected serially to provide printouts of results.

The PDP 11/03 is a 16 bit word computer. It consists of an LSI 11/02 microprocessor with an KEV-11 floating point chip, an BDV-11 bootstrap and diagnostic board, an DLV-11 4-port serial interface board, and an DRV-11 parallel interface board. The operating system is RT-11 version 04B. Programming is done in FORTRAN IV version 2.5. The FORTRAN software is also provided by Digital Equipment Corporation.

III. Data Processing

The analysis of Mossbauer spectra makes use of several characteristics of the Mossbauer effect. These characteristics are called parameters and the parameters have the same relationships regardless of the absorber used. These parameters will determine the relative intensity of each line to the other lines, the half-width at half-maximum of the dip corresponding to a line, and the velocity position of the line relative to zero velocity. A computer program to analyze the spectra must be able to use these parameters to compute a theoretical spectrum that most nearly matches the experimental data.

Mossbauer Parameters

The parameters that are used to compute the theoretical spectrum (or fitting the spectrum) are divided into three groups: intensity, position, and half-width. If the energy distribution is not truly a Lorentzian, a fourth set of parameters called form may be required.

The intensity of a line is defined as the number of counts between the lowest portion of a dip and the baseline. The baseline is the horizontal line across the top of the spectrum that represents the spectrum from the source photons if none were absorbed (see Figure 1). For example, if the baseline contained 100000 counts and the lowest part of a dip contained 80000 counts, the intensity of that line would

be 20000 counts. This requires that the baseline and the intensity of each line be variable parameters. Intensities of some lines are related so that one intensity parameter may be used for more than one line. The intensity of a line may also be expressed as an area, which is the intensity multiplied by the half-width at half-maximum. In some absorbers (natural iron for example), the areas of two lines may be closer related than the intensity and the area may be the best parameter to use.

Since the intensity depends strongly on accurately determining the baseline, some additional parameters may be required. In reality, the baseline is not flat, but has a slight curvature caused by the varying source-to-absorber distance. The closer the source is to the absorber, the higher the number of counts in the baseline. This difference is caused by the increased solid angle the absorber represents when the source is nearer. This curvature may be handled in several ways including polynomial fits to the baseline, allowing the geometry to be a variable, or assuming the baseline is a straight line. If the spectrum to be fit is a full period spectrum, there is a point of symmetry in the spectrum. This symmetric point, called the fold point, represents the point of inflection in the baseline curvature. The baseline is concave in one half of the spectrum and convex in the other. If the spectrum is folded in half at this point, the baseline curvature will tend to cancel out and the dips in each of the mirror spectra will

coincide, increasing the statistics of each dip. The curvature and the fold point must be considered when determining parameters to use to fit Mossbauer spectra.

The position of a line relative to zero velocity (and thus relative to the unshifted source photon energy) is determined by some combination of three parameters; isomer shift, hyperfine magnetic splitting, and quadrupole coupling. The isomer shift is the amount the midpoint of the spectrum is shifted from zero velocity. The isomer shift is caused by the difference in the nuclear radii of the ground and excited states, and how this difference couples with the electron charge density at the nucleus. The hyperfine magnetic splitting results from a coupling of the nuclear magnetic moment with the magnetic field due to the atom's own as well as adjacent atom's electrons. The result will be pairs of lines corresponding to transitions between nuclear magnetic sublevels with the lines in each pair separated by an amount proportional to the magnetic field present. Quadrupole coupling results from the interaction of the nuclear quadrupole moment with the gradient of the electric field produced by other charges in the crystal. The quadrupole coupling also results in a splitting of lines. These effects and the Hamiltonians used to represent them are described in detail by Roberts (Ref 5:5). The position of a line may be determined by a combination of these effects but the combination is consistent for a particular type of absorber.

The half-width parameter is the half-width of a dip at

half the maximum intensity of the dip. It is measured in velocity units and is related to the Lorentzian lineshape of the dip.

The form parameters are only used if the distribution function to be used is not pure Lorentzian. These parameters may be used to change the lineshape from Lorentzian to Gaussian or to some combination of both to account for fluctuations in the energy distribution of a line.

Program GENFIT

The computer program GENFIT is a least-squares minimization curve fitting program developed at Argonne National Laboratory. This has been the computer program used in the past to analyze Mossbauer spectra. The design and memory requirements of GENFIT require a mainframe computer and, therefore, this program could not be used on the PDP 11 microcomputer. It is, however, used as a reference to compare with the results obtained using MOSFUN and the PDP 11 system.

GENFIT uses a user-supplied subroutine called CALFUN to relate the Mossbauer parameters to the spectrum from a particular absorber. This subroutine is the model used by GENFIT to find the best fit to the experimental data. GENFIT assumes a quadratic curvature to the baseline and then subtracts this curvature as background from the baseline. This leaves a linear baseline for intensity calculations. The shape, relative intensities, position, and half-width of the lines is determined by the model described in the CALFUN

subroutine to be written for each absorber type used. In addition, no provisions exist to modify the model during an evaluation. GENFIT is described in more detail by Skluzacek (Ref 8:16).

MOSFUN Background

The computer program used for this study is a FORTRAN IV program called MOSFUN. It was written by E.W. Muller at the Johannes Gutenburg-Universitat in Mainz, West Germany and is available through the Mossbauer Effect Data Center, University of North Carolina (Ref 4:90).

MOSFUN is written in a very general manner so that it can be used on a variety of computers. It is available in three formats: batch, console (interactive), and a modified console version for use on PDP systems. Since the program is written very generally, there are no machine-specific functions used and no special spectrometer capabilities required.

Four models for use in fitting spectra are provided, with simple provisions for expanding to more complex models. Two models (called theories by MOSFUN) are done using Lorentzian lineshapes, one using dip intensity and one using dip area. The other two theories use approximate Voigt lineshapes, again one using dip intensity and one using dip area. Iterations can be carried out using either Newton's method or the Gradient method to determine the corrections required in the theoretical spectrum.

MOSFUN uses the geometry of the baseline to account

for its curvature and is capable of analyzing both full period and half period spectra. The parameters are read from a disk file rather than requiring new subroutines to change absorbers. The parameters can be modified during the fit process if one of the interactive versions is used.

The PDP version of MOSFUN is used for this study with some modifications to better use the current spectrometer. The PDP version is basically the console version (designed to run on mainframe computers using terminals) reduced in size, so that it can be overlaid into the PDP computer's smaller memory. MOSFUN makes extensive use of subroutines and this makes overlaying much easier. MOSFUN contains 17 subroutines that are divided into two overlay regions. The PDP version allows up to 32 fit parameters, 24 lines, and 1024 data points (Ref 3:4.14). Three of the subroutines (DATI, DRIVE and PLOT) were extensively modified, as was the main program in the areas that used these subroutines. Installation of MOSFUN on the PDP 11/03 is described in Appendix A.

Main Program

The main program for MOSFUN (Ref 3:3.3) acts primarily as an interactive guide. One of 16 commands (Ref 3:4.16) is entered from the console, then the main program determines what initial values are needed and which subroutines are to be called. Initial values are requested and entered from the terminal. Data and parameter values are stored on disk files and are entered from the disk. Variables are transferred between the subroutines and the main program through common

blocks. Output can be directed to the terminal or the line-printer.

The PDP 11 uses a technique called overlaying to allow programs much larger than the computer's memory to execute. In this process, more than one subroutine can share the same computer memory space as long as both are not needed at the same time. MOSFUN takes advantage of this capability by placing all subroutines that execute a command in one overlay region. Any subroutine that these subroutines may need to call is placed in a second overlay region. If more than one subroutine is needed at the same time, one may be placed in the main root segment which is always in the computer's memory. A breakdown of the overlay regions and the sub-routines that each contains is:

Region	Subroutines			
Main Root	MOSFUN	DRIVE	FTHEO	
Overlay Region 1	DATI	PARM	TMIN	SAVE
	FIT	NORM	SHIFT	SUBT
	PRNT	PUNC	PLOT	
Overlay Region 2	TRANS	LINEQ	THEO1	THEO2

When a command is given, the main program, MOSFUN, calls the appropriate subroutine from overlay region 1. These subroutines handle the experimental data, the parameters, the correlation matrix, the fit procedure, spectrum manipulation, and the output of results. The subroutines in overlay region 2 enter any values into the program from the console, solve a

system of linear equations, and model the theoretical spectrum. Each command is entered interactively from the terminal by typing the first two letters of the command. The commands may be given in any order after the experimental data is entered and the parameter starting values described.

Subroutine Descriptions

A brief description will be given concerning the function of each subroutine. Each subroutine and its variables and indicators are described in detail in the MOSFUN documentation (Ref 3:3.1-3.27), so will only be reviewed here. The modifications made to the program will be covered in more detail. The word "channel" describes the address of a data point within a series of storage locations. This may be in an MCA or a computer memory chip.

Subroutine DATI (Ref 3:3.8) handles the input of experimental data from the disk and sets the channel limits over which the spectrum will be fit. The experimental data is checked point by point for dropped counts and DATI provides the opportunity to zero any channel that appears bad. A bad data point is one that differs from the previous channel by more than statistical error times a correlation factor entered from the console. Any channel that has been zeroed is ignored for fit calculations and is not plotted. Any channel outside the fit limits is set negative and is ignored for fit calculations but is plotted. Overflows are added to the data during this subroutine.

DATI was modified to identify and sort out counts

from the internal time-keeping oscillator and the Moire interferometer for velocity measurement which are multiplexed with the data by the spectrometer. The counts, proportional to velocity when extracted, are corrected for overflows, converted to a velocity, and related to a channel number by a linear least squares fit. These data points are then set to zero so that the remaining data are identified as spectral data. Appendix B contains a listing of the changes to DATI.

Subroutine DRIVE (Ref 3:3.22-3.26.3) calculates the velocity and geometry factor for a particular channel. It can also calculate the derivatives for parameters 1 (baseline), 2 (geometry), and 3 (folding point) which are used during the fit process. DRIVE calculations depend on the drive mode used with the transducer. Triangular wave and sine wave modes are provided in MOSFUN. The geometry factor accounts for the curvature of baseline caused by the varying source-to-absorber distance. The fold point represents the point of symmetry in a full period spectrum such as that generated by a triangular wave or sine wave. The velocity in any channel is determined by the maximum velocity (VMAX) relative to the channel's position in the period.

DRIVE was modified to calibrate the velocity relative to a channel from the velocity data extracted during DATI. This velocity fit is used to calculate the velocity for any channel. A third drive mode was added since all current work is done using the flyback mode. The flyback spectrum is essentially the right half of a triangular wave spectrum and

computations for geometry were carried out as such. The fold point is not used for flyback spectra since flyback is only a half period spectrum. Even though the fold point is a parameter it does not enter into the calculation of half period theoretical spectra. Appendix C contains a listing of the changes to DRIVE.

Subroutine PARM (Ref 3:3.12-3.13) controls the input of, and changes to, the fit parameters. Each parameter is defined by an index number, a name, a value, and upper and lower limits. Once defined, the number is the only quantity required to reference that parameter. The upper and lower limits are bounds which restrict the values the parameter can assume. If the parameter exceeds either limit, the parameter is set equal to that limit for the next iteration and a warning message is displayed. If the limit is set equal to zero, the program is free to adjust the parameter to any value. Each parameter can be set to vary (FREE) or can be treated as a constant (FIX). Any time a command can be given, a parameter can be fixed or freed. This allows control over well defined parameters during the first few blocks of iterations. Four values of the parameters are available. The current values of these parameters can be changed to the starting values, the optimum values (the lowest Chi-squared), or the values of the previous iteration. The parameters must be replaced as a set. This would be done if divergence occurs or if convergence to a local minimum is occurring.

Subroutine TMIN (Ref 3:3.9-3.10) controls the correla-

tion matrix. This matrix is not a correlation matrix in the usual sense of error correlations, but rather correlates the lines in a spectrum to the parameters being used to fit that spectrum. This matrix determines how much of each parameter is used to calculate each line. It is actually four submatricies, one each for intensity, position, half-width, and form. Each submatrix has the dimensions of the total number of lines by the number of parameters that affect that submatrix. For example, if there are four lines in a spectrum and two intensity parameters, then the intensity submatrix will be a four by two matrix. The values in the correlation matrix are constants and do not vary during the fit of a spectrum.

Subroutine SAVE (Ref 3:3.19) enables the values of the parameters and the correlation matrix to be saved in one disk file. After any change is made to either set of values, the option to change the file (called the "save file") is offered. If changes are saved, the same save file can be used or another file created to save the values, if both sets are desired.

Subroutine NORM (Ref 3:3.14) performs a two parameter fit for the baseline and a scaling factor for the intensity. Relative intensities can then be used during the fit process to calculate real intensities and the baseline. Norm does not calculate Chi-squared or any theoretical information directly.

Subroutine FIT (Ref 3:3.16-3.17) initiates calculation

of Chi-squared and the theoretical spectrum. It performs iterations by either Newton's method or the Gradient method to calculate correction factors to be added to the parameters for the next test cycle.

Chi-squared is used as the test for convergence. It is normalized to the number of fitted channels and the number of free parameters. It is defined as (Ref 3:3.16):

$$\text{Chi-squared} = \left(\sum^{N} (Y(i)-F(i))^2 / Y(i) \right) / (N-M) \qquad (1)$$

where

N is the number of fitted channels

Y(i) is the i-th experimental data point

F(i) is the i-th theory value

M is the number of free parameters

When successive Chi-squared values differ by less than a specified convergence criteria (0.001 by default), iteration ceases and convergence is declared. If a Chi-squared value is larger than the previous value by more than ten times the convergence criteria, the theory is declared to diverge. Iteration stops regardless of the convergence status if the number of iterations specified by the user have been reached.

Newton's method (Ref 3:2.1-2.2) solves a system of equations (using subroutine LINEQ) created during the calculation of Chi-squared. The solutions represent the corrections to the fit parameters. Newton's method will diverge in the vicinity of flat minima such as those of the baseline. Therefore, care must be taken to use only the regions of the

spectrum that contain dips rather than fitting the entire spectrum.

The Gradient method (Ref 3:2.3-2.5) calculates the values of the corrections from the gradient of the previous values of the parameters. It works more slowly than Newton's method but will usually converge. A path width is specified which limits the amount the parameters will be changed in a single iteration. A large path width allows larger changes to be made in the parameter's value and may allow divergence if care is not taken. Too small a path width may delay convergence. The default value of the path width (1000) was used successfully in this study.

Subroutine FTHEO (Ref 3:3.27) resides in the main root region and directs which theory model will be used to calculate the theoretical spectrum. FTHEO is called by subroutine FIT and in turn calls the appropriate theory-model subroutine from overlay region 2 based on the theory index number selected during parameter entry.

Subroutine THEO1 (Ref 3:5.4-5.6) calculates the parameters using simple correlated Lorentzian curves. If a spectrum is composed of M(i) Lorentzians of intensity W(i), position E(i), and half-width H(i), then the theoretical value F at velocity v is given by (Ref 3:5.4):

$$F(v) = g\left(B - \sum_{i=1}^{M(i)} \left(W(i) \frac{1}{1+a^2}\right)\right) \qquad (2)$$

where

g is the geometry factor at velocity v

B is the baseline

a = (v-E(i)) / H(i)

The correlation matrix entered during TMIN determines how the parameters are combined to determine W(i), E(i), and H(i).

Subroutine THE02 (Ref 3:5.8-5.12) calculates the theoretical values using Voigt lineshapes. A Voigt profile is the result if statistical disturbances change the Lorentzian lineshape. A Voigt lineshape is a convolution of a Gaussian and a Lorentzian shape. This function cannot be analytically determined but can be approximated by the biquadrated (fourth power polynomial) function (Ref 3:5.8):

$$\frac{1}{1 + ax^2 + (1-a)x^4} \qquad (3)$$

A simple Lorentzian is produced when a=1 and a=0.5 approximates a Gaussian; thus "a" is constrained to be between 1 and 0.5. The theoretical value F at velocity v for a spectrum of M(i) lines with intensity W(i), position E(i), half-width H(i), and form factor A(i) is (Ref 3:5.11):

$$F(v) = g \left(B - \sum_{i=1}^{M(i)} W(i) \frac{1}{X2} \right) \qquad (4)$$

where

X2 = 1 + (A(i) X1^2) + ((1-A(i)) X1^4)

X1 = (v-E(i)) / H(i)

g = geometry factor at velocity v

B = baseline

The line parameters W, E, H, and A again are correlated by the correlation matrix as in THE01 except a submatrix is added for the form factors.

Subroutine SUBT (Ref 3:3.14) subtracts out the intensity contribution from any line by subtracting the theoretical fit for that line from the original data. This routine may be useful to remove strong well-defined peaks that may be hiding other peaks. The revised spectrum may be plotted or punched for examination, however, the original spectrum is restored before any additional fitting or testing is done.

Subroutine SHIFT (Ref 3:3.18) shifts a full period spectrum to correct for phase errors in the drive system. This places the zero velocity point directly in the center of a folded spectrum. For example, if the zero velocity channel of a folded spectrum containing 400 data points is in channel 204, then SHIFT will adjust the entire spectrum to put the zero velocity in channel 200.

Subroutine TRANS (Ref 3:3.20) allows format-free reading of numbers and is called anywhere in the program where numerical input is required. One input record (up to 80 columns) is entered from the terminal and transferred to TRANS via common blocks. This record can contain up to 16 values in any format separated by a blank. Each number is deciphered separately and then transferred back via common blocks in the vector OUT. The variable IOUT counts how many numbers have been analyzed and is transferred back in the subroutine statement.

Subroutine LINEQ (Ref 3:3.27) solves a system of linear equations by the Cholesky method. The Cholesky method works fast and requires no memory space for provisional results. This method is only applicable to symmetrical positive definite matricies. However, that is all that is generated during Newton's method or during the least squares fit of the velocity data extracted by subroutine DATI. LINEQ can also calculate the inverse of a matrix when it is needed, such as to calculate the parameter error correlations.

Subroutine PRNT (Ref 3:3.15) sends the results to the lineprinter. Options that may be printed are:

(1) Original data

(2) Parameters, parameter errors, and error correlations

(3) Output from the theory program that gives the intensity, position, half width, and form for each line

Subroutine PLOT is written to plot the experimental data and the theoretical spectrum, if it has been calculated. The plot routine provided in MOSFUN will only work with a VT-55 graphics terminal. Since this system does not have that option, a new alphanumeric plot routine was written. The plot can be directed to the terminal or the lineprinter. Intensities are scaled to the baseline and the plot can be done between any two channels. Appendix D contains a listing of subroutine PLOT.

Subroutine PUNC (Ref 3:3.18) stores the experimental and, if calculated, the theoretical spectra in a disk file for fitting in a later run. The spectra are written to the disk

under a file name requested from the terminal. Subspectra can also be calculated for each intensity parameter and their values are written to the disk separately.

Analysis Commands

The analysis of a spectrum requires several commands, and the order in which they are executed is somewhat flexible. Data, parameter values, and the correlation matrix must be entered, initial testing of the parameter set performed, spectrum folded or shifted if desired and then the fit procedure begun. Results may be printed, plotted, or saved any time a command can be given. Entering commands requires only the first two letters of the command be typed from the terminal. Commands and their prompts are discussed in detail in chapter 4 of the MOSFUN documentation (Ref 3:4.15-4.32).

Entering data is accomplished by the command DATIN. The storage file for the data and a correlation factor are requested. A correlation factor of 100 is sufficient to check for dropped counts in most spectra. The format for data on the disk is explained in Appendix E. The data is loaded into the computer's memory and a velocity calibration done. Any points that appear bad are listed with an option to zero. The option to zero any range of channels and/or set a range of channels that will not be fit is offered next. Any time a command can be given, DATIN may be commanded and the fit limits changed, without changing the experimental data.

The command PARIN allows entering of a parameter set, changing of any parameters value, and saving of the current

values on the disk. Appendix E contains a detailed explaination of how to enter a new set of parameters.

The command CORR enables initial entry of the correlation matrix or changes to the current values. If the correlation matrix is read from a file, it is part of the same file as the parameters and is read during the command PARIN. Details of the correlation matrix entry are described in Appendix E.

The command NORM should be given after entering the data parameters. This enables the program to use relative intensities rather than actual intensities and baseline.

The command TEST will perform an initial Chi-squared calculation on the initial parameters. If the Chi-squared value is too large, the parameters may need to be changed before the fit is begun.

The commands FOLD and SHIFT modify the experimental data to account for a full period spectrum. These commands would be used before the fit process is started. However, all spectra analyzed during this study were only half period spectra and these commands were not needed.

The fit procedure is begun by issuing the command NEWTON or GRADIENT. Both will request the number of iterations desired and the convergence criterion. A default convergence criterion of 0.001 is used if none is specified. For the GRADIENT command a path width is also requested. The default value of 1000 was used during this study. Larger values may allow faster convergence since larger changes can be made to the parameters, however, larger values may also lead to divergence. Small values

may needlessly delay convergence but may be needed if the default value leads to divergence. The number of iterations is kept small initially and the parameters evaluated after each iteration. Changes in the parameters can be made if problems begin to appear during the iteration process.

The commands RESTORE, FIX, and FREE allow control over the parameters. RESTORE changes the current parameter set to either the starting set, the optimum set, or the set from the previous iteration. These changes are made to the entire set of parameters. Selective changes must be made using the command PARIN. Commands FIX and FREE allow any combination of parameters to be fixed as constants or allowed to vary. Fixing well-defined parameters early insures major adjustments are only made to weaker parameters. Freeing only well-defined parameters allows the fit to be made to these parameters, and then the weaker parameters can be selectively released.

After each iteration, the following values are displayed on the terminal: the current value and previous value of each parameter, the Chi-squared value for the current set of parameters, the difference between this Chi-squared and the previous one, and the percent of the theory points that are within a statistical error's range of the experimental value. Statistical error is defined as the square root of the experimental value.

The commands PRINT and PLOT send the results to the lineprinter. PRINT can print the original data, the parameters alone or with their respective errors and error corre-

lations, and each line's value for intensity, position, half-width, and form from the theory program. PRINT also asks for text of up to 60 characters that can be used to describe the conditions that led to the results. PLOT can generate an alphanumeric plot and send it to either the terminal or the lineprinter. The plot is made between two specified channels and includes original data and the theoretical spectrum, if it has been calculated. The velocity scale is printed on the X-axis and relative intensity on the Y-axis.

The command PUNCH initiates transmission of the theory spectrum or subspectra for each intensity parameter to an output file on a disk. This file can then be used for fitting on a later run.

The command SUBTRACT removes the contribution from any theoretical intensity parameter from the experimental spectrum. This may reveal hidden peaks that are buried under much more intense peaks. After the subtraction, the resulting spectrum can be plotted or punched. Before any other command is executed, the original spectrum is restored by the program.

The command EXIT is the last command given, as it terminates program execution. A new analysis can be started without exiting by giving the commands DATIN and PARIN. Any files that are open are closed before execution is terminated.

IV. Results

To test the capabilities of MOSFUN to analyze spectra, four absorbers were chosen as test samples: potassium ferrocyanide, natural iron, sodium nitroprusside, and neodymium cobalt. Potassium ferrocyanide presents a simple, one-line spectrum with the location of the dip determined only by the isomer shift. Natural iron presents a six-line spectrum with the isomer shift and hyperfine splitting determining the position of the lines. Sodium nitroprusside has a two-line spectrum with the isomer shift and quadrupole coupling determining the position of the lines. Neodymium cobalt has a complex spectrum with several overlapping six-line iron spectra. All three position parameters are involved in determining the position of each set of lines and the parameters will be different for each set. Spectra for the first three absorbers were taken during the summer of 1981 using the current MCA based Mossbauer spectrometer. The neodymium cobalt spectrum was taken in 1976 by John (Ref 9:117) but used basically the same system. The data was manually stored on a computer disk where it could be recalled for analysis. The data for sodium nitroprusside and natural iron were also transferred to punched cards and analyzed by GENFIT on the mainframe computer for comparison with MOSFUN.

General Procedure

A standard procedure was used to analyze each set of data. The data was obtained and put into the computer. A set

of parameters was chosen based on known quantities for each sample. The data was then analyzed. The results obtained were compared to the results obtained with GENFIT and to published standards.

All analysis runs made during this study were half period spectra using the flyback drive mode. Each set of data was stored in 400 channels. Time-base data was stored in the first 14 channels and velocity counts multiplexed in every channel ending in a 9 (19,29,...,399). Mossbauer data was stored in all remaining channels.

After the data and parameter set were entered, commands TEST and NORM were executed. NEWTON was always tried, but always diverged before a final result could be found. Therefore, results shown were obtained using the Gradient method with the default convergence criterion and path width. Five iterations were tried initially and the parameters examined after each iteration. Then blocks of ten iterations were executed until convergence was reached. The number of iterations and the time required for each iteration depended on the number of Mossbauer lines and the number of free parameters.

Both Lorentzian and Voigt lineshapes were tried with each set of data. The final results reached were essentially the same, but convergence always took much longer using the Voigt lineshape. The similarity in the Lorentzian and Voigt results shows that there is very little statistical fluctuation in the experimental data.

The fit could be made over less than the full 400

channels. The range over which the data is to be fit is called the "fit range". In some cases, the fit range was varied to test the influence this had on convergence. No significant differences were noted in the final results; although convergence was generally faster if the fit was performed only over the range of channels containing the dips.

The results give an error limit for some parameters. The limits are somewhat deceiving between MOSFUN and GENFIT. MOSFUN and GENFIT both compute an error correlation matrix that tracks the propagation of system errors. GENFIT estimates the value of the standard deviation of a parameter as the square root of the diagonal element of this error matrix that corresponds to that parameter. The GENFIT error is for 67% confidence so is just this standard deviation. MOSFUN estimates the standard deviation as the square root of the diagonal element multiplied by the Chi-squared value. MOSFUN computes errors on a 95% confidence level which is 1.96 times the standard deviation. Therefore, the MOSFUN errors appear larger, relative to GENFIT, than they really are.

Potassium Ferrocyanide

The potassium ferrocyanide crystal is a 1 inch diameter by 1/8 inch thick sample (number E26532). A seven hour run was made to obtain data for analysis. The data was analyzed using Lorentzian lineshapes with the intensity model. Results show an isomer shift relative to natural iron of -0.0304 mm/s compared to a reference value of -0.035 $\pm$ 0.007 mm/s (Ref 10: 154). The computed value is within the range of the standard

value. Chi-squared for this analysis was 2.045. The computed half-width was 0.127 mm/s. No other analysis was done with this sample.

Natural Iron

The natural iron absorber is a 25 micron thick foil provided with the cobalt-57 source. A four hour run was made to collect data and the data was analyzed using GENFIT. The CALFUN subroutine is called EAFECAL. The data was then evaluated using MOSFUN with the peak area as the intensity model (theory 2-Lorentzian and theory 4-Voigt). The areas of peaks 1, 2, and 3 were required to equal the areas os peaks 6, 5, and 4 respectively. Comparative results are shown in Table I. Figure 2 shows a representative plot of the fit obtained for peaks three and four (the inner two iron peaks) using the Lorentzian area as the intensity.

The results show a difference of less than 1% for all parameters except half-width. MOSFUN had a half-width that is 10% larger than GENFIT. The difference in the results obtained using the Voigt and Lorentzian lineshapes is negligible. However, the Voigt lineshape required 33 gradient iterations to converge compared to 7 for the Lorentzian.

All parameter values for either program are within 1% of the published standard values (Ref 10:154) for natural iron.

Sodium Nitroprusside

The sodium nitroprusside absorber is a National Bureau of Standards crystal measuring 1 cm by 1 cm by 0.0775 cm thick. A six hour spectrum was obtained and analyzed using

GENFIT. The CALFUN routine was NANPCAL. Analysis with MOSFUN was done using the peak intensity as the intensity parameter (theory 1 - Lorentzian and theory 3 - Voigt). Both peaks were required to have the same intensity and half-width. Results are presented in Table II, which compares the MOSFUN theories to GENFIT. Figure 3 shows a plot of the fit obtained using the Voigt lineshape.

The results show virtually no difference between GENFIT and either MOSFUN theory. Again the Voigt lineshape took longer than the Lorentzian to fit (41 gradient iterations to 22). Results for all methods are within 1% of the standard values (Ref 10:154).

Neodymium Cobalt

A neodymium cobalt-5 spectrum had been obtained in 1976 and was analyzed since it is a more complicated spectrum than any of the other samples used. Neodymium cobalt was the subject of Skluzacek's thesis. This particular spectrum was taken two years later after some equipment changes. The preparation of the sample is described by Skluzacek (Ref 8: 12-13) and will not be repeated.

Analysis with MOSFUN was done using theory model 1, Lorentzian intensities, and modeled three of the four sites modeled by Skluzacek (Ref 8:40-41). The fourth site has only a minor impact. In addition, the source used to obtain this spectrum was very weak and this reduced the relative intensities of the peaks by a factor of two, causing the fourth site to be buried. In the analysis, each site was assumed to

produce a six line iron spectrum with the intensities of peaks 1 and 3 held at a 3 to 1 ratio. The intensity of peak 2 was free to adjust.

The results were inconclusive due to the very weak intensity of the inner lines. The weak source presented poor statistics for the inner peaks of the stronger two sites and all of the third site. The outer peaks for the two stronger sites were converged upon and gave results that agreed well with Skluzacek's results. The stronger sites had magnetic fields of 239.9 kOe (kilo Oersted) and 204.7 kOe compared to 239.3 kOe and 205.5 kOe respectively for Skluzacek (Ref 8: 38-39). All other parameters are tied to the location of the inner peaks and no unique values could be found.

Program Considerations

MOSFUN is a user-oriented program and as such is very easy to use. However, parameter selection and control will make a great deal of difference in the results obtained. A TEST should be made to insure the initial selection of parameters and the correlation matrix are reasonable. The number of initial iterations should be kept small (approximately 5) and the parameters examined closely. The use of parameter limits will also keep parameters from deviating too far from reasonable values. To keep the number of free parameters low, use the same parameter for more than one peak whenever possible.

Table I

Natural Iron Comparative Results : MOSFUN vs. GENFIT

Parameter	MOSFUN(Loren.)	MOSFUN(Voigt)	GENFIT
Chi-squared	2.15	2.35	2.33
Isomer Shift (mm/s)	-0.1054±.0024	-0.1055±.0024	-0.1057±.0005
Mag Field (kOe)	328.6 ±.4	328.5 ±.4	329.4 ±.1
Intensity Ratio			
peak 2 to 1	0.873 ±.014	0.875 ±.035	0.8708±.0098
peak 3 to 1	0.547 ±.086	0.549 ±.022	0.5451±.0088
Peak Separation(mm/s)			
peaks 1-6	10.571	10.571	10.578
peaks 2-5	6.121	6.123	6.125
peaks 3-4	1.671	1.669	1.672
Half-width (mm/s)			
peak 3	0.125 ±.006	0.124 ±.011	0.113 ±.005
peak 4	0.131 ±.007	0.131 ±.011	0.118 ±.011

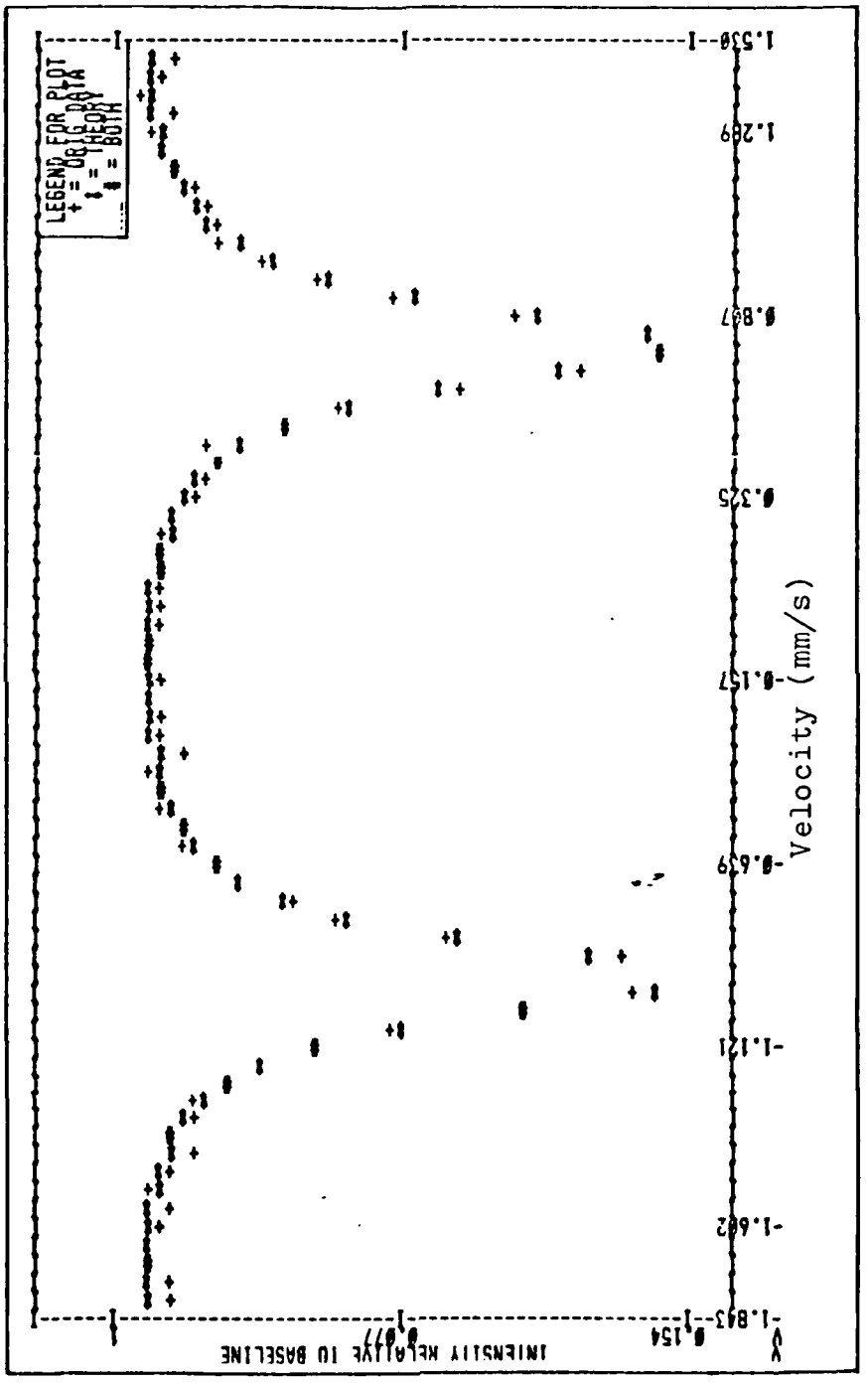


Figure 2. Natural Iron Results for Peaks 3 and 4
MOSFUN Using Lorentzian (Areas)

Table II

Sodium Nitroprusside Comparative Results : MOSFUN vs. GENFIT

Parameter	MOSFUN(Loren.)	MOSFUN(Voigt)	GENFIT
Chi-squared	1.502	1.349	1.111
Isomer Shift (mm/s)	-0.367 ±.029	-0.367 ±.046	-0.3665±.0006
Quadrupole Split(mm/s)	1.692 ±.115	1.692 ±.181	1.693 ±.001
Peak Location (mm/s)			
peak 1	-1.213	-1.213	-1.213
peak 2	0.479	0.479	0.480
Half-width (mm/s)	0.139 ±.010	0.139 ±.018	0.137 ±.002

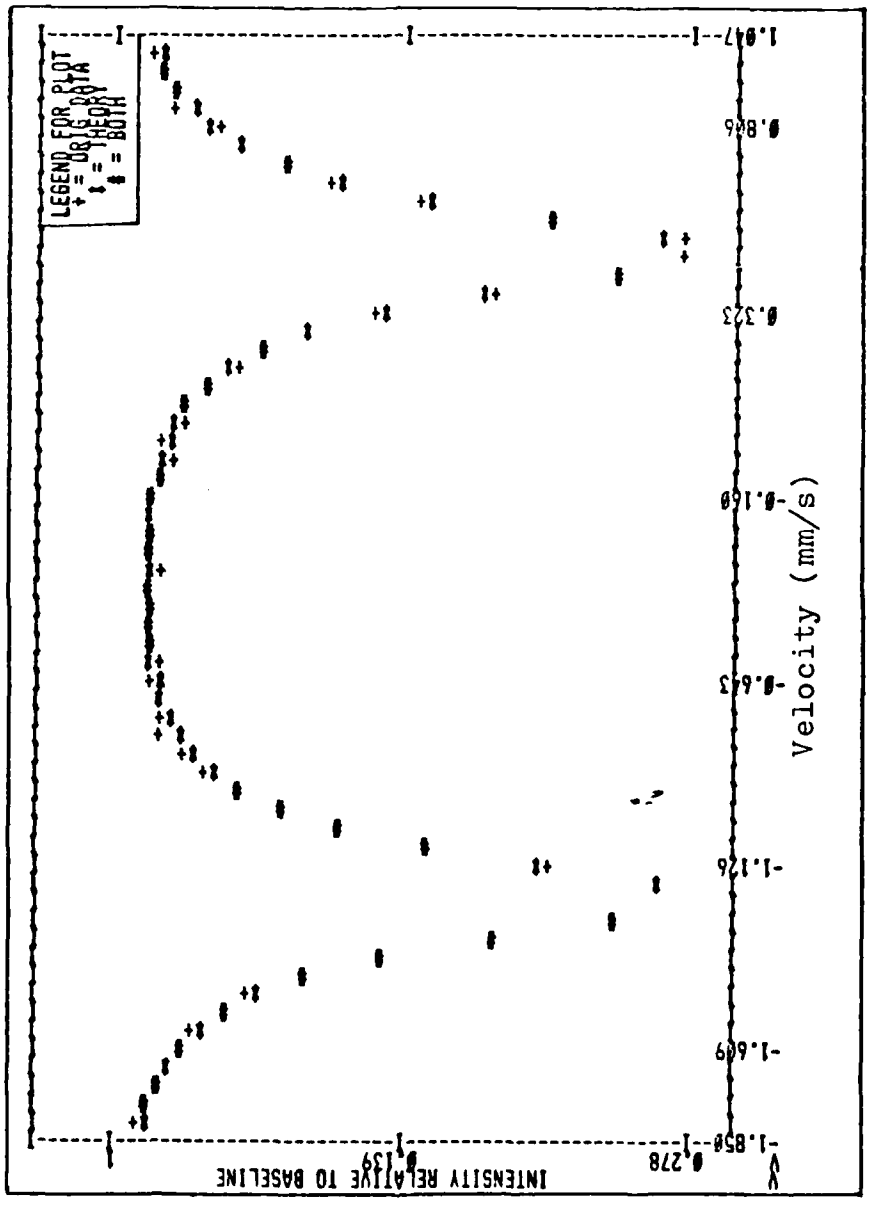


Figure 3. Sodium Nitroprusside Results
MOSFUN Using Voigt (Inten.)

V. Conclusions and Recommendations

Conclusions

The MOSFUN program greatly improves the analysis of Mossbauer spectra, even without the new hardware that is to be developed. MOSFUN is faster for the user, easier to control, more efficient due to this extra control, easier to expand, and gives results that agree within 1% with those obtained using GENFIT and a mainframe computer. MOSFUN will be able to analyze the data collected by the new hardware, when that hardware has been completed.

Recommendations

The ease with which MOSFUN can be expanded and the parameters adjusted make this program the best to use for all future Mossbauer spectra evaluation. MOSFUN should be expanded to use the triangular wave mode and evaluated when triangular wave mode spectra can be generated. MOSFUN should be used to analyze spectra from the current MCA based system, even before the new hardware is developed. The data must be transferred to the disk manually, but this is still faster than the current evaluation procedure.

Bibliography

1. Beller, D.E. "Crystallization Kinetics of Two Metallic Glasses by Mossbauer Spectroscopy." Unpublished MS thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, March 1981.

2. Frauenfelder, Hans. The Mossbauer Effect. New York: W.A. Benjamin, Inc, 1962.

3. Muller, E.W. MOSFUN-A New and Versatile Mossbauer Fitting Program. Description and user's guide. Mainz, West Germany: Institut fur Anorganishe Chemie und Analytiche Chemie, Johannes Gutenburg-Universitat, 1980. [This document is available from the Mossbauer Effect Data Center]

4. Muller, E.W. "MOSFUN-A New and Versatile Mossbauer Fitting Program," Mossbauer Effect Reference and Data Journal, 4 (4): 89-90 (1 April 1981).

5. Roberts,L.D. "Mossbauer Studies of Metallic Glasses." Unpublished MS thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, December 1978.

6. RT-11 System User's Guide. User's guide, Maynard, Massachusetts: Digital Equipment Corporation, March 1981. (AA-5279B-TC)

7. Schmidt, T.A. "An Analysis of Metallic Glasses by Mossbauer Spectroscopy." Unpublished MS thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, March 1978.

8. Skluzacek, E.W. "Analysis of the Mossbauer Spectra of NdCo5." AFML-TR-75-162. Wright-Patterson AFB, Ohio: Air Force Materials Laboratory, 1976.

9. Skluzacek, E.W. Thesis Laboratory Notebook, Physics Department, Wright-Patterson AFB, Ohio, 1976.

10. Stevens, J.G. and Virginia E. Stevens. Mossbauer Effect Data Index: Covering the 1972 Literature. New York: IFI/PLENUM Data Corporation, 1973.

11. Towery, C.D. "Mossbauer Study of the Magnetocrystalline Anisotropy of NdCo5." Unpublished MS thesis. Wright-Patterson AFB, Ohio: School of Engineering, Air Force Institute of Technology, June 1971.

Appendix A

MOSFUN Installation Procedure

FORTRAN programs on the PDP 11 require three actions before they can be executed. Each FORTRAN routine is written and stored on the disk with a FOR file type. Each routine is compiled using the FORTRAN command (Ref 6:4-93) and an object file with an .OBJ file type created on the disk. Then the object files are linked together using the LINK command (Ref 6:4-119). LINK also adds any library routines that are needed. The final executable program is created with a .SAV file type. This is the only file needed for execution and is executed by a "RUN filename" command.

Three disk volumes are used to install and use MOSFUN. Volume 1 is the FORTRAN Working Volume and contains the FORTRAN compiler, operating system, and system library. It also contains the necessary utility programs to edit, compile, and link FORTRAN routines. Volume 2 is the MOSFUN Files Volume and contains all of the .FOR and .OBJ files to be linked to create MOSFUN.SAV. This volume is only used to edit and recompile MOSFUN if changes are made. Volume 3 is the MOSFUN Working Volume and contains MOSFUN.SAV and the data files necessary to analyze Mossbauer spectra using MOSFUN. This is the only volume needed to analyze spectra. Volumes 1 and 3 are bootable into the RT 11 single job operating system. Volume 2 is strictly a storage volume and must be used with one of the other volumes.

To edit a MOSFUN routine, put Volume 1 into drive DY0

and type "BOOT DY0:". This sets up the default values to edit and store routines on a second volume. Put Volume 2 into the other drive. Give the command "EDIT filename.FOR" and the appropriate FORTRAN routine will be put into the KED editor. After the changes have been made, compile the routine using the command "FORTRAN/NOLINENUMBER filename". The NOLINENUMBER option is required for MOSFUN to fit in the computer's memory. If errors exist and the line number is needed, the routine may be compiled with line numbers in order to correct errors. However, recompile without line numbers before linking. If a listing is desired, the option /LIST can be added to the FORTRAN command. If added before the routine name, the listing is sent to the lineprinter. If added after the routine name, the listing is stored on the disk with a .LST file type.

After a routine has been edited and successfully compiled, all the object files must be relinked. During the link procedure, the overlay regions are set up. To make the link procedure easier, an indirect command file "LNKMOS.COM" was set up. This file executes the commands needed to link all of the object files into the correct overlay regions and create MOSFUN.SAV. To link MOSFUN, give the command "LINK@LNKMOS". One major problem will frequently appear when this is done. Volume 2 is filled nearly to capacity. If there are not 300 contiguous blocks on the disk, a "SAV DEVICE FULL" error is generated due to the size of MOSFUN.SAV. If this occurs, delete the old .SAV file using the command "DELETE MOSFUN.SAV". Delete any backup text files (.BAK file types) and any other files that are not needed. Then squeeze the

remaining files so that all open blocks are at the end of the disk by giving the command "SQUEEZE DY1:". When this is completed try the link procedure again.

Once MOSFUN.SAV is ready, put Volume 3 into DY0 and boot it so that all default storage is to the MOSFUN Working Volume. Then copy MOSFUN.SAV using the command "COPY DY1: MOSFUN.SAV DY0:". MOSFUN is now ready to run again. Volumes 1 and 2 can be stored as they are no longer needed.

Appendix B

Subroutine DATI

Subroutine DATI controls the input of the experimental data from the disk. This subroutine was modified to identify and sort out counts from the internal time-keeping oscillator and the Moire interferometer for velocity measurement which are multiplexed with the experimental data by the spectrometer.

The identification text for the data, the format for the data, and several constant values are read from the file. The format for these inputs is described in Appendix E. If a format is not specified, a format of (8F10.0) is used for the experimental data. The first constant value was changed from the maximum velocity to the number of time-base overflows (TOVR). The next four constants still describe the number of data overflows (NOVF), the number of data points (N), the period or the number of channels a full period spectrum would contain (PERI), and the drive mode number (NDRV). A sixth and seventh constant were added to enter the number of channels between the channels containing Moire counts (MPLEX) and the first channel containing these counts (NST). All of these constants are transferred in common blocks, except NST which is transferred in the subroutine statement.

The time-base counts are read from the first 13 channels, overflows added, and then averaged so that the variable TIME contains the average number of time-base oscillator counts. These channels and channel 14 are zeroed

so they will not be considered in the rest of the program. TIME is transferred to subroutine DRIVE in common block FLD.

The velocity data (Moire counts) is read from channels NST to PERI/2 in jumps of MPLEX. Thus only the first half period is used for velocity calibration. If the spectrum is only a half period spectrum, then the entire spectrum is used since PERI/2 should equal N. The velocity points are checked for overflows, overflows are added, and the left half of the velocity curve made negative. These counts are then transferred to the array VCON. VCON(1) contains the number of velocity points (maximum of 64). VCON(2) to VCON(VCON(1)) contain the velocity data in counts which can be related to channel number in subroutine DRIVE.

A listing of the changed portion of subroutine DATI is attached. DATI after label 250 is unchanged from the original version.

Listing of Subroutine DATI

```
      SUBROUTINE DATI (N1,N2,N3)
C
      COMMON /UNIT/ NRD,NWR,NCR,LPR,LFD,NPL,NVS,MAXN,MAXM,MAXF,NV(1211
      COMMON /DAT/  VMAX,PERI,Y(1024),ID(40),NDRV,N,NF,VEL(10)
      COMMON /THEO/ F,P(32),DF(32),NTEO,M,MF,KTEO(20),KP(32)
      COMMON /FLD/  P3,B,V,VCON(65),V1,V2,GEO,B0,C0,NFLD,MSH,MPLEX,TIN
      COMMON /TRA/  IN(81),OUT(16)
C
      DIMENSION KFRM(8),IFRM(8),INST(4)
      DATA IFRM/2H(1,2H0F,2H8.,2H0),2H  ,2H  ,2H  ,2H  /
      DATA INST(1)/1HL/,INST(2)/1HZ/,INST(3)/1HH/,INST(4)/1H*/
      DATA NY/1HY/
C
C     N1 = INPUT UNIT
C     N2 = FACTOR FOR SEARCHING DROPPED CHANNELS
C     N3.NE.0 ERROR RETURN
C
      N9=N2
      IF(N1.EQ.0) GOTO 285
      READ(N1,5200,END=202)(ID(I),I=1,40)
      READ(N1,5200,END=202)(KFRM(I),I=1,8)
      READ(N1,5210,END=202)(IN(I),I=1,80)
      GOTO 203
C
C     EOF
C
  202 WRITE(NWR,6205) N1
      N=0
      N3=1
      RETURN
  203 CALL TRANS(NN)
      TOVR=OUT(1)
      NOVF=OUT(2)
      N=OUT(3)
      PERI=OUT(4)
      NDRV=OUT(5)
      MPLEX=OUT(6)
      NST=OUT(7)
  205 IF (N.EQ.0) N=400
C
C     MORE THAN MAXN POINTS
C
      IF(N.LE.MAXN) GOTO 220
      N=0
      N3=1
      WRITE(NWR,6200) MAXN
      RETURN
  220 IF(NDRV.EQ.0) NDRV=3
      IF(PERI.EQ.0.) PERI=N*2.
```

```
      IF(NST.EQ.0) NST=19
      P3=PERI * .5 + .5
 230  IF(KFRM(1).NE.IFRM(8)) GOTO 240
      DO 231 I=1,8
 231  KFRM(I)=IFRM(I)
 240  WRITE(NWR,6230) (ID(I),I=1,39)
      WRITE(NWR,6240) TOVR,NOVF,N,PERI,NDRV,KFRM
      DO 241 I=1,MAXN
 241  Y(I)=0.
      READ(N1,KFRM,END=250) (Y(I),I=1,N)
C
C     CALCULATE AVERAGE TIME OSC. COUNTS
C
  99  TIME=0.
      DO 100 I=1,13
      TIME=TIME+Y(I)+1000000.*TOVR
 100  Y(I)=0.
      TIME=TIME/13.
      Y(14)=0.
C
C     EXTRACT VELOCITY DATA, ACCOUNT FOR
C     OVERFLOWS AND ZERO AFTER PUTTING IN VCON.
C
      IIII=0
      III=0
      NEND=PERI/2
      DO 101 I=19,NEND,MPLEX
      VJMP=Y(I+MPLEX)-Y(I)
      IF(VJMP.GE.500000.) III=III+1
 101  CONTINUE
      J=1
      DO 102 I=19,NEND,MPLEX
      J=J+1
      IF(III.EQ.0) GOTO 105
      VJMP=Y(I+MPLEX)-Y(I)
      IF(J.GT.65) GOTO 102
      VCON(J)=Y(I)+FLOAT(III)*1000000.
      IF (VJMP.GE.500000.) III=III-1
      GOTO 102
 105  VJMP=Y(I)-Y(I+MPLEX)
      IF(J.GT.65) GOTO 102
      VCON(J)=Y(I)+FLOAT(IIII)*1000000.
      IF(VJMP.GE.500000.) IIII=IIII+1
 102  Y(I)=0.
      Y(399)=0.
      III=-1
      DO 106 I=3,J-1
      VJMP=ABS(VCON(I-1))-VCON(I)
      IF(ABS(VCON(I)-VCON(I+1)).GT.(2.*VJMP)) III=1
      VCON(I)=VCON(I)*FLOAT(III)
 106  CONTINUE
      VCON(2)=-VCON(2)
      VCON(1)=J
```

```
            NN=N1
            NK=N2
            N1=0
            N2=NST
            CALL DRIVE(N1,N2)
            N1=NN
            N2=NK
            GOTO 260
C
C           LESS THAN N DATA IN PARTITION N1
C
 250        WRITE(NWR,6210) I,N
```

Appendix C

Subroutine DRIVE

Subroutine DRIVE calculates the velocity, geometry factor, and derivatives of the first three parameters (baseline, geometry, and fold point) for any channel. The original subroutine DRIVE used the maximum velocity and the period to determine the velocity contribution to these values. Subroutine DRIVE was changed to allow a linear least squares fit to be performed on the velocity data extracted by subroutine DATI. This velocity fit is then used to find the above values.

Two arrays are built to use during the velocity fit. Array X(I) contains the channel number corresponding to the velocity in VCON(I). Array XY(I) contains VCON(I) which has been converted from counts to millimeters per second (mm/s) by the formula:

$$XY(I) = 317.5 \frac{VCON(I)}{TIME} \qquad (C1)$$

The range of array X is scaled to a range of +1 to -1. The sums of powers of X (XDP(1-3)) and the sums of XY times the powers of X (XYDP(1-2)) are calculated. The values of XDP are placed in a symmetric array A and the values of XYDP placed in vector B. Subroutine is called to solve the system of linear equations AZ=B. The results (Z) and the range of X are scaled back to the original values. The velocity calibration is stored in the vector VEL. The velocity (V) in any channel (X) can then be calculated by the formula:

$$V(mm/s) = VEL(1) * X + VEL(2) \qquad (C2)$$

A listing of subroutine DRIVE is attached. Only the flyback mode calibration is shown, since it was the only drive mode used during this study.

Listing of Subroutine DRIVE

```
      SUBROUTINE DRIVE (N1,N2)
      COMMON /UNIT/ NRD,NWR,NCR,LPR,LFD,NPL,NVS,MAXN,MAXM,MAXF,NV(12
      COMMON /DAT/  VMAX,PERI,Y(1024),ID(40),NDRV,N,NF,VEL(10)
      COMMON /THEO/ F,P(32),DF(32),NTEO,M,MF,KTEO(20),KP(32)
      COMMON /FLD/  P3,B,V,VCON(65),V1,V2,GEO,B0,C0,NFLD,MSH,MPLEX,T
      COMMON /TRA/  IN(81),OUT(16)
      COMMON /LIN/  BB(32),A(32,32)
C
      DIMENSION X(65),XY(65),XDP(10),XYDP(10)
C
C     N1 = 0  CALCULATE VELOCITY COEFF.
C     N1 = 1  VELOCITY V FROM CHANNEL NUMBER XX
C     N1 = 2  V,GEO
C     N1 = 3  DERIVATIVES P(1)-P(3)
C     N1 = 4  DERIVATIVES P(1)-P(2)
C
C     N2 = 1st CHANNEL CONTAINING VELOCITY (N1=0)
C     N2 = CHANNEL NUMBER   (FOR N1 = 1-4)
C     N2 = 0  ERROR RETURN
C
C     NDRV = 1  TRIANGULAR WAVE MODE
C     NDRV = 2  SINE WAVE MODE
C     NDRV = 3  FLYBACK WAVE MODE
C
      IF(N1.NE.0) GOTO 1000
      NST=N2
      NFN=VCON(1)
      GOTO (1,2,3), NDRV
    1 WRITE(NWR,6000)
      RETURN
    2 WRITE(NWR,6200)
      VCON(1)=0.
      RETURN
    3 WRITE(NWR,6300)
      X(2)=NST
      XY(2)=VCON(2)*156.25/TIME
      DO 401 I=3,NFN
      X(I)=X(I-1)+MPLEX
  401 XY(I)=VCON(I)*156.25/TIME
      L=2
      K=1
C
C     FIND MIN AND MAX VALUES FOR X
C
      XMIN=X(2)
      XMAX=X(2)
      DO 402 I=3,NFN
      XMIN=AMIN1(XMIN,X(I))
  402 XMAX=AMAX1(XMAX,X(I))
```

```
C
C       ZERO ARRAYS FOR SUMMING
C
        MM=2*K+1
        DO 403 I=1,MM
        XDP(I)=0.
        XYDP(I)=0.
 403    CONTINUE
C
C       TRANSFORM RANGE OF X TO (-1,1) AND COMPUTE SUMS OF
C       POWERS OF X AND SUMS OF XY TIMES POWERS OF X.
C
        C1=2.0/(XMAX-XMIN)
        C2=(XMAX+XMIN)/(XMAX-XMIN)
        LU=2*K+1
        LL=K+2
        DO 404 II=2,NFN
        XP=1.
        XI=C1*X(II)-C2
        DO 405 I=1,L
        XDP(I)=XDP(I)+XP
        XYDP(I)=XYDP(I)+XP*XY(II)
 405    XP=XP*XI
        DO 404 I=LL,LU
        XDP(I)=XDP(I)+XP
 404    XP=XP*XI
        DO 406 I=1,L
        BB(I)=XYDP(I)
        DO 406 J=1,L
        A(I,J)=XDP(I+J-1)
 406    CONTINUE
C
C       CALL LINEQ AND FIT VELOCITY DATA
C
        N1=0
        N2=2
        CALL LINEQ(N1,N2)
C
C       MOVE VELOCITY CALIB. TO VEL ARRAY
C
        DO 407 I=1,L
        JK=K-I+2
 407    VEL(JK)=BB(I)
C
C       ADJUST COEFF. TO ORIGINAL RANGE OF X
C
        DO 408 I=1,K
        DO 408 J=1,I
 408    VEL(J)=VEL(J)*C1
        C1=(XMAX+XMIN)/2.0
        DO 409 I=1,K
        MM=L-I+1
        DO 409 J=2,MM
```

```
 409     VEL(J)=-C1*VEL(J-1)+VEL(J)
C
C        COMPUTE MAXIMUM AND ROOT MEAN SQUARE
C        ERRORS AND OUTPUT ERROR ANALYSIS
C
         WRITE(NWR,6400)
         EMAX=0.
         SUM=0.
         VMAX=0.
         DO 410 I=2,NFN
         YC=VEL(1)
         DO 411 J=1,K
 411     YC=YC*X(I)+VEL(J+1)
         VMAX=AMAX1(VMAX,ABS(YC))
         DIFF=YC-XY(I)
         IF (I-1.GT.L) GOTO 413
         WRITE(NWR,6410) I,X(I),XY(I),YC,DIFF,VEL(I-1)
         GOTO 412
 413     WRITE(NWR,6420) I,X(I),XY(I),YC,DIFF
 412     EMAX=AMAX1(EMAX,ABS(DIFF))
 410     SUM=SUM+DIFF**2
         ERMS=SQRT(SUM/FLOAT(NFN-1))
         WRITE(NWR,6430) EMAX,ERMS
         RETURN
C
 1000    GOTO (10,100,200),NDRV
C
C        FLYBACK MODE
C
 200     IF(N1.GE.3) GOTO 250
         XX=N2
         PER=PERI/4.
         IF(XX.LT.PERI/2.) GOTO 232
         XX=XX-PERI
         C1=P3-PER-XX
         S=1.
         GOTO 240
 232     C1=P3+PER-XX
         S=-1.
 240     C4=S*VMAX/PER
         V=VEL(1)*XX+VEL(2)
         IF(N1.EQ.1) RETURN
         CX=V/VMAX
         C2=S*(CX*CX-1.)
         C5=1.+P(2)*C2
         GEO=1./C5/C5
         RETURN
C
C        DERIVATIVES
C
 250     MM=KP(1)
         IF(MM.GT.0) DF(1)=GEO
         MM=KP(2)
```

```
      IF(MM.GT.Ø) DF(MM)=-2.*GEO*BØ*C2/C5
      IF(N1.EQ.4) RETURN
      MM=KP(3)
      IF(MM.GT.Ø) DF(MM)=GEO*((V/XX)*CØ-BØ*P(2)*V*4./C5/VMAX/PER)
      RETURN
C
 5ØØØ FORMAT(8ØA1)
 63ØØ FORMAT(1H ,4X,'DRIVE MODE FLYBACK WAVE')
 64ØØ FORMAT(1H ,//,1ØX,36HPOLYNOMIAL LEAST SQUARE FIT
     1ANALYSIS,//,4H    I,6X,7HX-GIVEN,7X,7HY-GIVEN,6X,
     28HY-FITTED,8X,5HERROR,1ØX,6HVEL(I),//)
 641Ø FORMAT(1H ,I3,4X,4(1PE1Ø.3,4X),1PE13.6)
 642Ø FORMAT(1H ,I3,4X,4(1PE1Ø.3,4X))
 643Ø FORMAT(1H ,9X,5HEMAX=,1PE15.6,9X,5HERMS=,1PE15.6)
      END
```

Appendix D

Subroutine PLOT

Subroutine PLOT performs an alphanumeric plot on the terminal or the lineprinter. The plot will always contain the original data, but will also include the theoretical spectrum, if it has been calculated. The plot is performed over a range of channels specified by two parameters, V1 and V2. If these are zero, the default range will plot between channel 20 and the last channel (N).

The Y-axis represents intensity scaled relative to the baseline, so that the most intense dip within the channel range is full scale. The X-axis represents velocity in mm/s.

The plot is done as a one line text vector. Integer variables are used to calculate the relative intensity of the data point or the theory value, and scale them between 1 and 50. A symbol (+ for original data, * for theory, and # for both) is placed in the text vector at the corresponding value in the vector, and all other points are blank. This text vector is then printed out. Every tenth channel also has the velocity value printed on the X-axis.

A listing of subroutine PLOT is attached.

Listing of Subroutine PLOT

```
      SUBROUTINE PLOT (N1,N2)
      COMMON /UNIT/ NRD,NWR,NCR,LPR,LFD,NPL,NVS,MAXN,MAXM,MAXF,NV(12
      COMMON /DAT/  VMAX,PERI,Y(1024),ID(40),NDRV,N,NF,VEL(10)
      COMMON /THEO/ F,P(32),DF(32),NTEO,M,MF,KTEO(20),KP(32)
      COMMON /FDAT/ G,GP,GOPT,GOLD,GAM,YY(1024)
      COMMON /FLD/  P3,B,V,VCON(65),V1,V2,GEO,B0,C0,NFLD,MSH,MPLEX,T
      COMMON /TRA/  IN(81),OUT(16)
C
C     ***************************************************************
C     N1 = 6  LPR PLOT
C     N1 = 7  CONSOLE PLOT
C
C     N2 = 0  YY (THEORY and ORIG DATA)
C     N2 = 1  Y  (ORIG DATA)
C     ***************************************************************
C
C     N3,N4  CONSOLE PLOT FROM CHANNEL N3 TO CHANNEL N4
C
      DIMENSION NTXT(58)
      DATA NBL/1H /,NSTAR/1H*/,NPLUS/1H+/,NPND/1H#/
      K1=N1
      K2=N2
      N3=20
      N4=N
      IF(V1.NE.0.)N3=V1
      IF(V2.NE.0.)N4=V2
      IF(N3.LE.0) N3=20
      IF(N4.GT.N) N4=N
      WRITE(K1,6705)ID
      IF(KTEO(1).NE.NBL) WRITE(K1,6706)KTEO
      WRITE(K1,6707)N3,N4
      WRITE(K1,6700)(ID(J),J=1,6)
C
C     SCALE Y AXIS
C
      BASE=Y(N3)
      IF(P(1).NE.0.)BASE=P(1)
      YMAX=0.
      YMIN=1.E10
      DO 1200 I=N3,N4
      G1=Y(I)
      IF(N2.EQ.0) G1=YY(I)
      IF(G1.EQ.0.) GOTO 1200
      YMAX=AMAX1(YMAX,G1)
      YMIN=AMIN1(YMIN,G1)
 1200 CONTINUE
      G2=YMAX-YMIN
      IF(BASE.LE.G2) BASE=YMAX
      WRITE(K1,6710) G2/BASE,.5*G2/BASE
```

```
C
C       BEGIN PLOT
C
        N1=1
        N2=N3
        CALL DRIVE(N1,N2)
        WRITE(K1,6720) V
        K3=N3+1
        K4=N4-1
        DO 1250 I=K3,K4
        G1=Y(I)
        G6=YY(I)
        NQ=INT((G1-YMIN)/G2*50)
        MQ=INT((G6-YMIN)/G2*50)
        IF(G1.LE.0.) NQ=0
        IF(G6.LE.0.) MQ=0
        DO 1210 J=1,58
        NTXT(J)=NBL
        IF(J.EQ.NQ) NTXT(J)=NPLUS
        IF(J.EQ.MQ) NTXT(J)=NSTAR
        IF(J.EQ.MQ.AND.J.EQ.NQ) NTXT(J)=NPND
 1210   CONTINUE
        IF(MOD(I,10).NE.0) GOTO 1220
        N1=1
        N2=I
        CALL DRIVE(N1,N2)
        WRITE(K1,6730) V,NTXT
        GOTO 1250
 1220   WRITE(K1,6740) NTXT
 1250   CONTINUE
        N1=1
        N2=N4
        CALL DRIVE(N1,N2)
        WRITE(K1,6720) V
        RETURN
C
 6700   FORMAT(1H ,20X,'MOSFUN PLOT OF ',15X,'LEGEND FOR PLOT',/,22X,6(
        118X,'+ = ORIG DATA',/,' VEL.',48X,'* = THEORY',/,'(MM/S)',48X,
        2'# = BOTH',/,'  V',20X,'INTENSITY RELATIVE TO BASELINE')
 6705   FORMAT(1H ,//,5X,40A2)
 6706   FORMAT(1H ,'THEORY= ',10A2,5X,10A2)
 6707   FORMAT(1H ,/,15X,'CHANNEL ',I4,' TO CHANNEL ',I4)
 6710   FORMAT(1H ,' V',4X,F7.3,19X,F7.3,22X,1H1)
 6720   FORMAT(1H ,F7.3,'--I------------------------I
        1------------------------I------I')
 6730   FORMAT(1H ,F7.3,3X,58A1,1HI)
 6740   FORMAT(1H ,5X,1HI,4X,58A1,1HI)
        END
```

Appendix E

MOSFUN User Supplement

This appendix contains supplemental information on how to enter the initial values required for MOSFUN to operate. This information adds to the MOSFUN documentation and is necessary due to the changes made to DRIVE. It also expands the information on the entry of the parameters and the correlation matrix.

MOSFUN requires three sets of input before a spectrum can be evaluated. These are the experimental data, the parameter set, and the correlation matrix. Examples of these inputs are attached. Example 1 is an experimental data disk file for sodium nitroprusside (SNP), example 2 is the interactive sequence to enter the parameter file for SNP, and example 3 is the interactive procedure to enter the correlation matrix for SNP. In examples 2 and 3, the computer printed entries are preceeded by the identifier "CMPTR:". All other entries would be typed in by the user. These examples may be referred to during the procedure descriptions to follow.

The experimental data is entered from the disk along with several other values. These entries are an identification text for the data, the format by which the data is to be read, and constants for the number of time-base overflows (TOVR), the number of data overflows (NOVF), the total number of channels to be read (N), the number of channels in a full period spectrum (PERI), the number representing the drive

mode (NDRV), the multiplex interval for the velocity data (MPLEX), and the first channel containing velocity data.

The parameter set is entered from the disk or the console. Each parameter is composed of an identifying number, a name, a value, and upper and lower limits. Also entered with the parameters are the theory number and the total number of parameters.

The correlation matrix is entered from the disk or the console. The correlation matrix relates the parameters to the lines that make up the spectrum. Entries made with the correlation matrix are descriptive text to identify the correlation matrix, the total number of lines in the spectrum, the number of parameters that will be used to determine the intensity, position, half-width, and form of the lines, and how much of each parameter is to be applied to each line.

Data

The data inputs are stored on the disk. They are read by MOSFUN when the command "DATIN" is given. MOSFUN asks for the location and file name for the data by the prompt "DYX:FILE.EXT". The file name (FILE) can have up to six characters but the file type (EXT) must have three characters. For example, to enter the sodium nitroprusside data from drive DY0, the response would be "DY0:SNP827.EXT".

A correlation factor will be requested by the prompt "CORR. FACTOR". The factor is entered from the console and is used to determine bad data points. A factor of 100 is sufficient for most sets of data.

The data is stored on the disk in any format, but the other information must preceed it in three records. Each record can contain up to 80 characters and is ended by a carriage return. The first record contains text to identify the data. The second record contains the FORTRAN format that MOSFUN will use to read the data. The format must be inclosed in parentheses. The default format is (8F10.0) and is selected by a blank second record. The third record contains the constant values required to describe the data. The values are separated by commas and are entered in the following order:

1. TOVR - the number of time-base overflows
2. NOVF - the number of data overflows
3. N - the total number of channels to be read (default = 400)
4. PERI - the number of channels in a full period (default = N)
5. NDRV - the number representing the drive mode
6. MPLEX- the multiplex interval for the Moire counts
7. NST - the first channel containing Moire counts (default = 19)

Default values can be selected by entering a zero or leaving the field blank.

Parameter Set

The parameter set may be read from the disk or the console. Parameter entry is started by the command "PARIN" from the console. MOSFUN prompts "INPUT FROM FILE (Y/N)" to ask whether the input is coming from a disk file or the

terminal. If the parameters are to be read from a disk file, MOSFUN asks for the file location and name in the same way it did for the data. However, the parameter file name must be six characters long. If the parameters are to be entered manually or changes are to be made to the current parameter set, MOSFUN begins an interactive session (see example 2).

The example uses sodium nitroprusside and uses nine parameters. Besides the required three baseline parameters, the parameters are the intensity for each of the two sodium nitroprusside lines, the isomer shift, the quadrupole coupling, and the half-width of each line.

Parameters must be entered from the console initially. MOSFUN prompts with the current value, changes are made, and the changed values are restated. The process continues until stopped by the user. The first input is the theory number and the number of parameters. MOSFUN will prompt "THEORY 1 , 0 PARAMETERS". The theory number selected and the number of parameters are entered, separated by a comma. Each parameter's values are entered in a like manner. MOSFUN will prompt with the number and name, such as "NR 1, NAME BASE (*=END)". A number and a name are required initially to define the parameter, but only the number is needed after it is defined. The program will then prompt with the current values for that parameter number in the form "VALUE 0.0000, LLIM 0, ULIM 0". The correct value and limits are entered, separated by commas. If no limits are entered, a value of zero is assigned and the parameter can vary without limits. If no values are entered, the previous values remain unchanged. All changed values are

restated by MOSFUN for review. The process continues until an astrisk is entered for a parameter number.

MOSFUN will then prompt "CHANGE SAVE FILE(Y/N)", to ask if the parameter set is to be saved to the disk. If not, PARIN stops. If yes, the program will ask which disk drive and what file name is to be used to store the file with the prompt "DYX:PAROUT.PAR". The response will state the drive, the file name,and the file type, and must contain 14 characters as in the prompt. The next prompt, "FILE NEW OR OLD", asks if the file is new or already on the disk. The file is then saved to the disk.

Correlation Matrix

The correlation matrix is read from the disk with the parameter set on the command "PARIN". To enter the correlation matrix from the console or to change any values, the command "CORR" is entered from the console, and an interactive procedure such as that in example 3 is started.

The first prompt asks where the final correlation matrix values are to be printed. This can be to the lineprinter, the console, or not at all. This printout will be done after all values or changes have been made but before the changes are saved.

The first input to the correlation matrix file is a short (20 characters) text entry that describes the correlation matrix. MOSFUN prompts with "THEORY (20 CHAR)".

The next input to the file is five integers that tell the number of lines and the number of parameters that

describe the intensity, position, width, and form of the lines. MOSFUN prompts with the current values of these integers, and changes are made by entering the new values, separated by commas. All five must be entered, even if only one is changed. A blank input leaves all values unchanged.

The example used is sodium nitroprusside and has two of each type of parameter; intensity, position, and half-width. The intensity of each peak is allowed to vary independently as are the respective half-widths. The position of line 1 is the isomer shift less one-half of the quadrupole coupling. The position of line 2 is the isomer shift plus one-half of the quadrupole coupling. There are no form parameters since a Lorentzian lineshape is assumed.

The correlation matrix is entered next as four submatricies, one each for intensity, position, width, and form. Each has one row for each line in the spectrum and one column for each parameter that affects that submatrix. For example, if there are six lines and four intensity parameters, then the intensity submatrix will have six rows and four columns. MOSFUN tells which submatrix is to be entered by the prompt "PARAMETER TYPE I (I, P, G, X, *=END)", where I is intensity, P is position, G is width, and X is form. Any submatrix can be selected by entering the appropriate letter. If the prompted entry is correct, a blank entry will begin the procedure to read the submatrix values. MOSFUN tells the column to be changed by a prompt that relates the submatrix column to a parameter. The prompt " 1. COLUMN 4. PARAMETER NAME (*=END)" means that column 1 relates parameter 4, called NAME, to the

lines in the spectrum. To change the column or parameter, enter the correct column number and the corresponding parameter number separated by a comma. A blank entry leaves the values unchanged. The values in that column are then listed, and changes can be made to all values in the column or to any row in the column. The values relate how many times that parameter will apply to each line. Once changes are made, an astrisk must be placed in the first column of an input to each of the above procedures.

Once all of the columns have been entered, the entire correlation matrix is printed as earlier directed. The prompt "CHANGE SAVE FILE (Y/N)" again appears and is exactly the same as for the parameter set, since both are stored in the same disk file.

Example 1. SNP Data Input File

```
SNP(8-27-81)  SODIUM NITROPRUSSIDE

5.0,0,400,800,3,10
  557102  556975  557036  556968  557036  557058  556941  556996  557076  556989
  556978  557065  556978  168526  113106  113203  113694  113592  316401  113550
  113193  113216  113382  113254  113981  113079  113183  112830  298433  113523
  113597  113546  113513  113255  113438  113667  113764  113178  281824  113116
  113251  113068  113174  113549  112774  113385  113458  113172  265269  113320
  112939  113135  113036  112953  113223  113861  113031  112738  247683  112383
  113001  113101  112674  112217  113591  112437  113079  112768  230960  113213
  112538  113052  112519  113105  113436  113215  113226  112919  213998  113104
  113184  112865  113469  112378  113314  113364  112734  112787  196889  112938
  112578  112983  112555  112611  112623  113663  112809  112845  179618  113079
  112717  112506  112486  112809  113373  112785  112763  112544  162369  112951
  112886  112229  112783  112609  112760  113093  112279  112923  144780  112768
  112475  112940  112312  112953  112396  112598  112674  112594  127628  112437
  112759  112203  112323  112920  113491  113883  113271  112707  110903  112397
  112856  112814  112331  112261  112678  112820  112372  112009  93370  112778
  112544  112458  112287  111393  112031  112281  111557  112316  76035  112388
  112037  111446  111998  110862  111593  111450  111222  110199  59446  108923
  108592  106309  105167  101910  97619  91408  84279  79871  42160  88437
  95275  100570  103483  106098  107892  108989  110162  110054  25011  111205
  110566  111043  110971  111137  111073  110620  111228  111021  8121  110724
  110617  109949  110156  109316  109303  108100  106191  104924  10734  98517
  91581  84673  88527  81037  88621  96133  100920  103413  25917  107300
  109432  109418  110250  110901  110970  111012  110369  111398  45317  111436
  111733  111776  111471  112079  111999  111912  111480  112200  61886  112133
  112420  112310  112386  112906  112209  112012  112262  112687  78160  112764
  112963  112509  112665  111829  112386  112674  112714  112786  96035  112140
  112775  112381  112765  113000  112709  112654  112789  113040  112677  112510
  112491  112748  112732  112827  112179  112861  112323  112546  130801  112958
  112881  113030  112924  112508  112629  113006  112893  113054  147547  112328
  113090  112301  112450  112840  112594  112189  112873  112564  164496  112580
  113121  112711  113489  112620  113532  113002  113023  113366  102313  113070
  113200  112889  112184  113348  112763  112680  112864  112882  199656  112855
  112700  113055  112962  113085  112075  112548  113131  112866  216300  112997
  112972  113144  113037  113043  112283  113204  113156  112907  233574  113024
  113573  113480  113327  113100  113057  113189  113057  113613  250489  113625
  113847  112638  113344  113281  113335  113273  113448  113351  267473  112535
  113147  113153  113980  113522  112741  113354  113117  113624  284430  113320
  112893  113944  112662  113041  113604  113197  113676  113710  300843  113165
  113731  112910  113155  113096  113663  113717  112810  113106  318098  113895
  113455  113724  113936  113627  113072  113853  113363  113132  334632  113930
```

EXAMPLE 2: Entering Parameter File

```
CMPTR:  *****COMMAND*****
        PA
CMPTR:  *****PA*****
CMPTR:  INPUT FROM FILE (Y/N)
        N

CMPTR:  THEORY 1    Ø PARAMETERS
        1,9

CMPTR:  THEORY 1  LORENTZIANS (INTEN.)    9 PARAMETERS
CMPTR:  NR 1    NAME BASE (*=END)
        hit carriage return (cr)

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        123456

CMPTR:     1    BASE    123456.ØØØØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 2    NAME GEO   (*=END)
        cr

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        .ØØ1

CMPTR:     2    GEO           Ø.ØØ1ØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 3    NAME FOLDP   (*=END)
        cr

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        Ø

CMPTR:     3    FOLDP         Ø.ØØØØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 4    NAME          (*=END)
        4 INTEN1

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        32ØØØ

CMPTR:     4    INTEN1   32ØØØ.ØØØØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 5    NAME          (*=END)
        5 INTEN2

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        32ØØØ

CMPTR:     5    INTEN2   32ØØØ.ØØØØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 6    NAME          (*=END)
        6 ISOSFT

CMPTR:  VALUE      Ø.ØØØØØ,      LLIM    Ø.ØØØØØ,      ULIM    Ø.ØØØØØ
        -.367
```

```
CMPTR:      6    ISOSFT       -Ø.3670Ø      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 7     NAME          (*=END)
        7 QDSPL

CMPTR:  VALUE       Ø.ØØØØØ,       LLIM    Ø.ØØØØØ,       ULIM    Ø.ØØØØØ
        1.69

CMPTR:      7    QDSPL         1.69ØØØ      Ø.ØØØØØ      Ø.ØØØØØ
CMPTR:  NR 8     NAME          (*=END)
        8 HFWD1

CMPTR:  VALUE       Ø.ØØØØØ,       LLIM    Ø.ØØØØØ,       ULIM    Ø.ØØØØØ
        .15,.1,.25

CMPTR:      8    HFWD1         Ø.15ØØØ      Ø.1ØØØØ      Ø.25ØØØ
CMPTR:  NR 9     NAME          (*=END)
        9 HFWD2

CMPTR:  VALUE       Ø.ØØØØØ,       LLIM    Ø.ØØØØØ,       ULIM    Ø.ØØØØØ
        .15,.1,.25

CMPTR:      9    HFWD2         Ø.15ØØØ      Ø.1ØØØØ      Ø.25ØØØ
CMPTR:  NR 9     NAME    HFWD2 (*=END)
        *

CMPTR:  *****COMMAND*****
```

EXAMPLE 3 Entering Correlation Matrix

```
CMPTR:  *****COMMAND*****
        CO
CMPTR:  *****CO*****

CMPTR:  CONTROL PRINTOUT Ø=NO,  1=CONSOLE,  2=LINEPRINTER
        1

CMPTR:  THEORY (2Ø CHAR.)
        INTEN,HW FREE

CMPTR:  NO. OF LINES,INT.-,POS.-,WIDTH-,FURTHER PARAM.
CMPTR:       Ø   Ø   Ø   Ø   Ø
        2,2,2,2,Ø

CMPTR:  PARAMETER TYPE I (I,P,G,X,*=END)
        hit carriage return (cr)

CMPTR:     1. COLUMN     4. PARAMETER = INTEN1   (*=END)
        cr

CMPTR:     1. COLUMN     4. PARAMETER = INTEN1   (*=END)
CMPTR:       1       Ø.ØØØØØ
CMPTR:       2       Ø.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, *=NONE
        A

CMPTR:  VALUES (MAX 16/LINE, *=END)
        1,Ø
        *

CMPTR:       1      1.ØØØØØ
CMPTR:       2      Ø.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, *=END
        *

CMPTR:     2. COLUMN     5. PARAMETER = INTEN2   (*=END)
        cr

CMPTR:     2. COLUMN     5. PARAMETER = INTEN2   (*=END)
CMPTR:       1      Ø.ØØØØØ
CMPTR:       2      Ø.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        A

CMPTR:  VALUES (MAX 16/LINE, *=END)
        Ø,1
        *
```

```
CMPTR:        1      0.00000
CMPTR:        2      1.00000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        *

CMPTR:  PARAMETER TYPE P (I,P,G,X,*=END)
        cr

CMPTR:     1. COLUMN     6. PARAMETER = ISOSFT   (*=END)
        cr

CMPTR:     1. COLUMN     6. PARAMETER = ISOSFT   (*=END)
CMPTR:        1      0.00000
CMPTR:        2      0.00000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        A

CMPTR:  VALUES (MAX 16/LINE, *=END)
        1,1
        *

CMPTR:        1      1.00000
CMPTR:        2      1.00000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        *

CMPTR:     2. COLUMN     7. PARAMETER = QDSPL    (*=END)
        cr

CMPTR:     2. COLUMN     7. PARAMETER = QDSPL    (*=END)
CMPTR:        1      0.00000
CMPTR:        2      0.00000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        A

CMPTR:  VALUES (MAX 16/LINE, *=END)
        -.5,.5
        *

CMPTR:        1     -0.50000
CMPTR:        2      0.50000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        *

CMPTR:  PARAMETER TYPE G (I,P,G,X,*=END)
        cr

CMPTR:     1. COLUMN     8. PARAMETER = HFWD1    (*=END)
        cr

CMPTR:     1. COLUMN     8. PARAMETER = HFWD1    (*=END)
CMPTR:        1      0.00000
CMPTR:        2      0.00000
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        C
```

```
CMPTR:  ROW NO., VALUE    (*=END)
        1,1
        2,Ø
        *

CMPTR:        1      1.ØØØØØ
CMPTR:        2      Ø.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        *

CMPTR:     2. COLUMN     9. PARAMETER = HFWD2    (*=END)
        cr

CMPTR:     2. COLUMN     9. PARAMETER = HFWD2    (*=END)
CMPTR:        1      Ø.ØØØØØ
CMPTR:        2      Ø.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        C

CMPTR:  ROW NO., VALUE    (*=END)
        2,1
        *

CMPTR:        1      Ø.ØØØØØ
CMPTR:        2      1.ØØØØØ
CMPTR:  A=CHANGE ALL ELEMENTS, C=SINGLE ELEMENT, (*=END)
        *

CMPTR:  PARAMETER TYPE X (I,P,G,X,*=END)
        *


CMPTR:  THEORY  INTEN,HW FREE
CMPTR:
CMPTR:          2 LINES
CMPTR:
CMPTR:          2 INTENSITY PARAMETERS
CMPTR:                   4          5
CMPTR:           1   1.ØØØØØ    Ø.ØØØØØ
CMPTR:           2   Ø.ØØØØØ    1.ØØØØØ
CMPTR:
CMPTR:          2 POSITION PARAMETERS
CMPTR:                   6          7
CMPTR:           1   1.ØØØØØ   -Ø.5ØØØØ
CMPTR:           2   1.ØØØØØ    Ø.5ØØØØ
CMPTR:
CMPTR:          2 HALF WIDTH PARAMETERS
CMPTR:                   8          9
CMPTR:           1   1.ØØØØØ    Ø.ØØØØØ
CMPTR:           2   Ø.ØØØØØ    1.ØØØØØ
CMPTR:
CMPTR:          Ø FURTHER PARAMETERS
CMPTR:
CMPTR:  CHANGE SAVE FILE (Y/N)
        Y
```

```
CMPTR:  DYX:PAROUT.PAR  (14 CHARACTERS)
        DYØ:SNP827.PAB

CMPTR:  FILE NEW OR OLD
        NEW

CMPTR:  ***** COMMAND *****
```

Vita

Bruce E. Pate was born on January 8, 1951 in Murfreesboro, Tennessee, and is the son of Gerald S. Pate and Donna J. Pate. He graduated from Quantico High School, Quantico, Virginia in June of 1969. In May of 1973, he graduated from Middle Tennessee State University, Murfreesboro, Tennessee with a Bachelor of Science Degree in Physics and Computer Science. He entered the Army through an ROTC commission in July of 1973. He has had duty at Fort Sheridan, Illinois and Aberdeen Proving Ground, Maryland as well as a tour in the Federal Republic of Germany. He and his wife, Kathy, have a daughter, Jenny, and a son, Mike.

Permanent Address: Route 7, Harrison Road
Murfreesboro, Tennessee
37130

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) Unclassified

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFIT/GNE/PH/82M-11 — 2. GOVT ACCESSION NO.: AD-A124 782	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Implementation of Software for a Microcomputer based Mossbauer Spectrometer	5. TYPE OF REPORT & PERIOD COVERED: MS Thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Bruce E. Pate	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE: March, 1982
	13. NUMBER OF PAGES: 73
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release: IAW AFR 190-17

LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (AIC)
Wright-Patterson AFB OH 45433

19 JAN 19[illegible]

~~Director of Information~~

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Mossbauer Spectroscopy
Computer Analysis
Microcomputer programming

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Computer analysis of Mossbauer spectra using a microcomputer requires tight controls on memory requirements and programming techniques. In this study, a FORTRAN IV program, MOSFUN, was programmed into an PDP-11 microcomputer. The program was modified to calculate the velocity for a data point using the least squares fit of counts from a Moire interferometer and to perform an alpha-numeric plot of the results and the original data. Spectra were

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

block 20 (cont)

obtained for absorbers of potassium ferrocyanide, natural iron, sodium nitroprusside, and neodymium cobalt. These spectra were then analyzed with the MOSFUN program and the results compared with analyses done with the former program, GENFIT, and with published values. Results showed MOSFUN to be more flexible, easier to adapt, and easier to control during the evaluation of spectra, while still providing results that agreed within 1% of those obtained with GENFIT and with standard values.

the previous values remain unchanged. All changed values are